

AMENDMENTS TO THE CLAIMS

1. (Previously presented) A computer-assisted method for translating a logic module interface, comprising the steps of:
 - determining a first interface of a logic module, the first interface having a first one or more interface element names; and
 - generating a second interface of the logic module, the second interface having a second one or more interface element names;
 - wherein the generating step comprises renaming the first one or more interface element names to produce the second one or more interface element names.
2. (Original) The method of Claim 1, wherein the renaming step comprises generating one, or more random strings as the second one or more interface element names.
3. (Original) The method of Claim 1, wherein the renaming step comprises hashing the first one or more interface element names.
4. (Original) The method of Claim 1, wherein the renaming step comprises adding an offset to the first one or more interface element names.
5. (Original) The method of Claim 1, wherein the generating step further comprises adding one or more dummy interface elements to the second interface.
6. (Original) The method of Claim 5, wherein the generating step further comprises associating one or more traps with the one or more dummy interface elements.
7. (Original) The method of Claim 1, wherein the logic module comprises an operating system and the first interface comprises a set of system calls of the operating system.

8. (Previously presented) A computer-assisted method for translating a user module, comprising the steps of:
 - determining a first user module and a translation, the translation mapping a first provider interface to a second provider interface; and
 - generating a second user module according to the translation;
 - wherein the generating step comprises replacing a first reference in the first user module to an element in the first provider interface with a second reference to an element in the second provider interface according to the translation.
9. (Original) The method of Claim 8, wherein the first provider interface comprises a set of system calls of an operating system, the second provider interface comprises a translated set of system calls of the operating system, the first user module comprises a software application referencing the first provider interface, and the second user module comprises a translated software application referencing the second provider interface.
10. (Previously presented) A computer system, comprising:
 - an operating system;
 - a processing module, wherein the processing module is running the operating system;
 - a first set of one or more system calls to the operating system for disclosing to trusted software modules; and
 - a second set of one or more dummy systems calls to the operating system for trapping untrusted software modules.

11. (Previously presented) A computer system comprising: a processing module; a software program; and an operating system; the software program configured for running on the operating system, the operating system having a translated system call interface, the translated system call interface comprising a first set of translated system call names for disclosing to trusted parties, the translated system call interface further comprising a second set of dummy system call names untranslating system call names for trapping entrusted parties, the software program comprising:
 - a set of one or more instructions;
 - wherein the set of one or more instructions comprises one or more references to the first set of translated system call names.
12. (Previously presented) A shared software library stored on machine readable media, comprising:
 - a first set of one or more library functions for disclosing to trusted software modules; and
 - a second set of one or more dummy library functions for trapping untrusted software modules.
13. (Previously presented) A hardware processing system, comprising:
 - a processor;
 - an instruction translation table; and
 - a module name translation table;
 - wherein the processor fetches an instruction, decodes the instruction according to a translation stored in the translation table, and executes the decoded instruction, and wherein instruction calls to modules are translated according to the module name translation table.

14. (Original) A method for generating a web page, comprising the steps of:
 receiving a web page request having a first URL;
 translating the first URL to a second URL according to a URL translation table,
 the second URL indicating a first web page, the first web page having a third one or more
 embedded URLs; and
 generating a second web page;
 wherein the generating step comprises replacing the third one or more embedded
 URLS with translated versions of said third one or more embedded URLs according to
 the URL translation table to obtain the second web page.
15. (Original) A method for processing a file access request, comprising the steps of:
 receiving a file access request having a first string; and
 translating the first string to a second string, the second string indicating a file
 name;
 wherein the translating step proceeds according to a file name translation table.
16. (Original) A method for providing access to a file system, comprising the steps of:
 receiving a file access request; and
 processing the request;
 wherein the file access request comprises a file name and a first file name
 extension, the processing step comprises translating the first file name extension to obtain
 a second file name extension and providing access to a file indicated by the file name and
 the second file name extension.

17. (Withdrawn) A method for processing a network connection request, comprising the steps of:
- receiving a network connection request having a first number; and
 - translating the first number to a second number, the second number indicating a port number;
 - wherein the translating step proceeds according to a port number translation table.
18. (Original) A method for processing a network packet, comprising the steps of:
- receiving a network packet, the network packet comprising a protocol type field having a first protocol type identifier;
 - translating the first protocol type identifier to a second protocol type identifier;
 - and
 - routing the network packet according to a protocol indicated by the second protocol type identifier;
 - wherein the translating step proceeds according to a protocol type field translation table.
19. (Currently Amended) A method for processing a database command, comprising the steps of:
- receiving a database command having a first command string; and
 - translating the first command string to a second command string, the second command string indicating a database command keyword;
 - wherein the translating step proceeds according to a database command keyword translation table.
20. (New) The computer system of claim 10, wherein the untrusted software module is disabled after a reference to any one or more of the second set of one or more dummy systems calls.
21. (New) The computer system of claim 10, wherein the operating system generates a system administrator alert after reference to any one or more of the second set of one or more dummy systems calls.

22. (New) The computer system of claim 10, wherein the operating system generates an event log after reference to any one or more of the second set of one or more dummy systems calls.
23. (New) The computer system of claim 11, wherein a second software program is disabled after the second software program references any one the second set of dummy system call name untranslated system call names.
24. (New) The computer system of claim 11, wherein the operating system generates a log event after a reference to any one or more of the second set of dummy system call name untranslated system call names.
25. (New) The computer system of claim 11, wherein the operating system generates a system administrator alert after reference to any one or more of the second set of dummy system call name untranslated system call names.
26. (New) The shared software library of claim 12, wherein a corresponding library function is disabled after a reference to the corresponding dummy library function.
27. (New) The shared software library of claim 12, wherein the untrusted software modules are disabled after a reference to the corresponding dummy library function.
28. (New) The shared software library of claim 12, wherein the dummy library function generates a system log event as result of a reference to the corresponding dummy library function.
29. (New) The shared software library of claim 12, wherein the dummy library function generates a system administrator alert as result of a reference to the corresponding dummy library function.
30. (New) The method of claim 14, wherein a reference any one or more third one or more embedded URLs before translation will generate a system administrator alert.